AN AUTOMATIC MULTIGRID METHOD FOR THE SOLUTION OF SPARSE LINEAR SYSTEMS

Yair Shapira
Mathematics Department, Technion -
Israel Institute of Technology, Haifa 32000, Israel.

Moshe Israeli and Avram Sidi
Computer Science Department, Technion -
Israel Institute of Technology, Haifa 32000, Israel.

SUMMARY

An automatic version of the multigrid method for the solution of linear systems arising from the discretization of elliptic PDE's is presented. This version is based on the structure of the algebraic system solely, and does not use the original partial differential operator. Numerical experiments show that for the Poisson equation the rate of convergence of our method is equal to that of classical multigrid methods. Moreover, the method is robust in the sense that its high rate of convergence is conserved for other classes of problems: non-symmetric, hyperbolic (even with closed characteristics) and problems on non-uniform grids. No double discretization or special treatment of sub-domains (e.g. boundaries) is needed. When supplemented with a vector extrapolation method, high rates of convergence are achieved also for anisotropic and discontinuous problems and also for indefinite Helmholtz equations. A new double discretization strategy is proposed for finite and spectral element schemes and is found better than known strategies.

1 INTRODUCTION

The multigrid method is a powerful tool for the solution of linear systems which arise from elliptic PDE's [1] [2]. This is an iterative method, in which the equation is first relaxed on the original fine grid in order to smooth the error; then the residual equation is sent to a coarser grid to be solved there and to supply a correction term. Recursion is used to solve the coarser grid problem in the same way the original equation is handled. In order to apply this procedure, the differential operator has to be discretized on all grids, and restriction and prolongation operators have to be defined in order to pass from coarse to fine grids and vice versa. The multigrid method works well for the Poisson equation in the square, but difficulties arise with non-symmetric problems, indefinite problems and problems with discontinuous coefficients or non-uniform grids. In those cases, it is not easy to discretize the differential operator on coarse grids and to generate the restriction and prolongation operators appropriately. Some suggestions about how to handle discontinuous coefficients are given in [4] and [5], while the singularly perturbed case is discussed in [6]. Slightly indefinite problems are discussed in [7]. These approaches involve special treatment of problems according to the original PDE, and the need for a uniform approach is not yet fulfilled.

In principle, the multigrid procedure is problem-dependent, and cannot serve as a "black box" that solves every problem. Special attention has to be given to the neighborhood of the boundary and to lines of discontinuity. In [8] [9] [10] an algebraic multigrid method for symmetric problems is developed. Though this method is automatic in the sense that it depends on the linear system of equations solely, it suffers from the disadvantage of the coefficient matrices for coarse grids being of 9-diagonal type, even when the original matrix is of 5-diagonal type [11]. An algebraic version of multigrid which overcomes this difficulty is presented in [12], and generalized to nonsymmetric problems in [13]. This version, however, does not improve the classical multigrid in cases of indefinite or hyperbolic problems and of non-uniform grids.

The algorithm which is presented in this work, and which we denote Multi Block Factorization (MBF) (the reason for this terminology will become clear in the next section), gives a uniform approach that enables one to handle the above difficulties. It relies on the algebraic system of equations solely, and not on the original PDE. The operators for coarse grids, as well as restriction and prolongation operators, are automatically defined when the coefficient matrix is given. It seems to be more robust than the classical multigrid method, as it solves non-symmetric problems (even hyperbolic or with closed characteristics) as quickly as classical multigrid solves the Poisson equation. Moreover, it is applicable to non-uniform grids as well, and does not require any special treatment of sub-domains. For anisotropic, discontinuous or indefinite problems MBF by itself is not always sufficient. However, it can cope with such problems successfully when it is applied in conjunction with vector extrapolation methods. In our numerical examples in the present work we have employed the Reduced Rank Extrapolation (RRE) of [17] and [18]. This and other related methods have been surveyed in [19] and their analysis provided in [20], [21] and [22]. The numerical implementation that we have used is the one given in [23].

The MBF algorithm is described in Section 2. In Section 3 numerical results are presented. In Section 4 the algorithm and the numerical results are discussed.

2 DESCRIPTION OF ALGORITHMS

2.1 Definition of the TBF Method

Let A be an $N \times N$ matrix. Let x and b be N-dimensional vectors. Consider the problem

$$Ax = b \tag{1}$$

An iteration of the Two-Block Factorization (TBF) method is defined by

$$\begin{aligned} TBF(x_{in}, A, b, x_{out}): & \\ x_0 &= x_{in} \\ x_{i+1} &= x_i - Z_i(Ax_i - b) \quad 0 \le i < i_1 \\ Q\bar{e} &= R(Ax_{i_1} - b) \\ x_{i_1+1} &= x_{i_1} - P\bar{e} \\ x_{i+1} &= x_i - Z_i(Ax_i - b) \quad i_1 < i \le i_1 + i_2 \\ x_{out} &= x_{i_1+i_2+1} \end{aligned} \tag{2}$$

where the Z_i are some preconditioning operators, i_1 and i_2 are nonnegative integers denoting the number of presmoothings and post-smoothings respectively and R, P and Q are operators to be defined later. Define

$$e_{in} \equiv x_{in} - x, \; e_{out} \equiv x_{out} - x$$

Then

$$e_{out} = [\prod_{i=i_1+1}^{i_1+i_2}(I - Z_i A)](I - PQ^{-1}RA)[\prod_{i=0}^{i_1-1}(I - Z_i A)]e_{in}$$

Consequently,

$$Q = RAP \Rightarrow e_{out} = 0 \Rightarrow x_{out} = x. \tag{3}$$

In the sequel, practical choices for Q will be considered. An iterative application of TBF is given by

$x_0 = 0,\ i = 0$
while $\|residual\| \geq \epsilon$
$\quad TBF(x_i, A, b, x_{i+1})$
$\quad i \leftarrow i + 1$
endwhile

2.2 Definition of the MBF Method

The Multi-Block Factorization (MBF) method is a modification of the TBF method, in which the system (2) is not solved directly, but is divided into several independent subsystems, which are solved directly or recursively by MBF itself. For simplicity, we first write the algorithm for tridiagonal systems. The operators P, R, Q and D will be defined later.

$MBF(x_{in}, A, b, x_{out})$:
if A is diagonal

$$x_{out} = A^{-1}b$$

otherwise:

$$\begin{aligned} D &\equiv diag(d_1, \ldots, d_N) \\ MBF & \quad (0, D^{-1}Q, D^{-1}R(Ax_{in} - b), \bar{e}) \\ x_{out} &= x_{in} - P\bar{e}. \end{aligned}$$

We turn now to the more general definition of MBF. First we note that if there exists a subset of coordinates of x which are independent of the others, then there exists a projection Π onto the sub-space spanned by those coordinates such that $(\Pi A \Pi)\Pi x = \Pi b$. In the following definition of MBF such sub-systems are solved directly, provided this can be done easily. The co-subsystem is solved recursively.

$MBF(x_{in}, A, b, x_{out})$:

1. If A is diagonal, set $x_{out} = A^{-1}b$ and stop.
2. If A includes an independent tridiagonal subsystem $(\Pi A \Pi)\Pi x = \Pi b$, solve it directly: $\Pi x_{out} = (\Pi A \Pi)^{-1}\Pi b$. If not, set $\Pi = 0$.
3.

$$\begin{aligned} y_0 &\equiv (I - \Pi)x_{in} \\ \bar{b} &\equiv (I - \Pi)b \\ y_{i+1} &= y_i - (I - \Pi)Z_i(Ay_i - \bar{b}) \quad 0 \leq i < i_1 \\ D &\equiv diag(d_1, \ldots, d_N) \\ MBF & \quad (0, D^{-1}Q, D^{-1}R(Ay_{i_1} - \bar{b}), \bar{e}) \end{aligned} \tag{4}$$

$$
\begin{aligned}
y_{i_1+1} &= y_{i_1} - (I-\Pi)P\bar{e} \\
y_{i+1} &= y_i - (I-\Pi)Z_i(Ay_i - \bar{b}) \quad i_1 < i \le i_1 + i_2 \\
(I-\Pi)x_{out} &= y_{i_1+i_2+1}.
\end{aligned}
$$

Trivially, one can replace action (4) by variant a :

$$
\begin{aligned}
MBF \qquad & (0, QD^{-1}, R(Ay_{i_1} - \bar{b}), e) \\
\bar{e} &= D^{-1}e
\end{aligned}
$$

or variant b :

$$
\begin{aligned}
MBF \qquad & (0, D^{-1/2}QD^{-1/2}, D^{-1/2}R(Ay_{i_1} - \bar{b}), e) \\
\bar{e} &= D^{-1/2}e.
\end{aligned}
$$

An iterative application of MBF is given by

$x_0 = 0,\ i = 0$
while $\|residual\| \ge \varepsilon$
$\qquad MBF(x_i, A, b, x_{i+1})$
$\qquad i \leftarrow i + 1$
endwhile

2.3 The Tridiagonal Case

Let I denote an identity operator. Suppose $N = 2^n$ for some positive integer n, and let A be a tridiagonal M-matrix satisfying $diag(A) = I$. Let $M(N)$ be the permutation matrix which reorders the variables of N-dimensional vectors such that odd numbered variables appear in a first block and even numbered variables appear in a second block. Define

$$M_0 = M(N),\ A_0 = A.$$

Then for some bidiagonal matrices B_0 and C_0 we have

$$A_0 = M_0^T \begin{pmatrix} I & B_0 \\ C_0 & I \end{pmatrix} M_0 = R_{A,1}^{-1} Q_{A,1} P_{A,1}^{-1},$$

where

$$R_{A,1} = \begin{pmatrix} I & 0 \\ -C_0 & I \end{pmatrix} M_0,\ Q_{A,1} = \begin{pmatrix} I & 0 \\ 0 & I - C_0B_0 \end{pmatrix},\ P_{A,1} = M_0^T \begin{pmatrix} I & -B_0 \\ 0 & I \end{pmatrix}.$$

Note that $Q_{A,1}$ is the Schur complement for A.

For $i = 1, 2, \ldots$, let I_i denote an identity operator of order $N - 2^{n-i}$. Let M_i be the $N \times N$ permutation matrix that reorders the coordinates x_i, $i = N - 2^{n-i} + 1, \ldots, N$, of an N-dimensional vector in the above manner, that is, order odd coordinates in a first block, then even coordinates in a second block. In fact,

$$M_i = \begin{pmatrix} I_i & 0 \\ 0 & M(2^{n-i}) \end{pmatrix}.$$

For $1 \leq i < n$, define

$$P_{A,i+1} = M_i^T \begin{pmatrix} I_i & 0 & 0 \\ 0 & I & -B_i \\ 0 & 0 & I \end{pmatrix}, \; R_{A,i+1} = \begin{pmatrix} I_i & 0 & 0 \\ 0 & I & 0 \\ 0 & -C_i & I \end{pmatrix} M_i,$$

$$Q_{A,i+1} = R_{A,i+1} A_i P_{A,i+1} = \begin{pmatrix} I_i & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I - C_i B_i \end{pmatrix},$$

$$D_{A,i+1} = diag(Q_{A,i+1}),$$

$$A_{i+1} = D_{A,i+1}^{-1} Q_{A,i+1} = M_{i+1}^T \begin{pmatrix} I_{i+1} & 0 & 0 \\ 0 & I & B_{i+1} \\ 0 & C_{i+1} & I \end{pmatrix} M_{i+1}, \quad (i < n-1).$$

The last equality sign implicitly defines B_{i+1} and C_{i+1}. For variants a and b, the above definition is modified to read

$$A_{i+1} = Q_{A,i+1} D_{A,i+1}^{-1}$$

and

$$A_{i+1} = D_{A,i+1}^{-1/2} Q_{A,i+1} D_{A,i+1}^{-1/2},$$

respectively.

Lemma 1 *For all variants, the matrices Q_i and A_i are tridiagonal M-matrices.*

Proof: The lemma follows from the definition by induction on i. □

Lemma 2 *The TBF method, when applied with*

$$Q \equiv Q_{A,1}, \; P \equiv P_{A,1}, \; R \equiv R_{A,1}, \; i_1 \equiv 0, \; i_2 \equiv 0$$

is a direct method.

Proof: Since

$$Q = Q_{A,1} = R_{A,1} A_0 P_{A,1} = RAP$$

the lemma follows from equation (3). □

The even numbered variables may be viewed as coarse-grid points. Then Q is a coarse grid operator, R is a fine-to-coarse restriction and P is a coarse-to-fine prolongation.

Lemma 3 *The MBF method applied with the operators*

$$A \equiv A_{i-1}, \; Q \equiv Q_{A,i}, \; D \equiv D_{A,i}, \; P \equiv P_{A,i}, \; R \equiv R_{A,i}$$

on the i^{th} call to the MBF procedure, is a direct method.

Proof: Note that on the $(n+1)^{st}$ call to the MBF procedure, the coefficient matrix A_n is diagonal, so the MBF procedure is a direct solve. By induction on $i = n, \ldots, 1$, all calls to MBF are equivalent to calls to TBF, hence are direct solves. □

In fact, in the tridiagonal case the MBF method is equivalent to the cyclic reduction method.

Note that if the matrices P and R are defined to be the rectangular matrices

$$P_{A,i+1} = M_i^T \begin{pmatrix} -B_i \\ I \end{pmatrix}, R_{A,i+1} = \begin{pmatrix} -C_i & I \end{pmatrix} M_i,$$

$$Q_{A,i+1} = R_{A,i+1} A_i P_{A,i+1} = I - C_i B_i,$$

$$D_{A,i+1} = diag(Q_{A,i+1}),$$

$$A_{i+1} = D_{A,i+1}^{-1} Q_{A,i+1} = M_{i+1}^T \begin{pmatrix} I & B_{i+1} \\ C_{i+1} & I \end{pmatrix} M_{i+1},$$

then the algorithm will still be applicable. As a matter of fact, the only difference between this method and the former is that in the present method we are not taking advantage of the known residuals on the odd numbered variables when making the coarse grid correction. Hence, if those residuals are zero, which may happen as a result of red-black presmoothing, the present method serves as a direct method, just as the former.

2.4 The Separable 2-Dimensional Case

Let $S \equiv (s_{i,j})$ and $T \equiv (t_{i,j})$ be matrices of order M and N respectively. Define

$$S \circ T = \begin{bmatrix} s_{1,1}T & \cdots & s_{1,M}T \\ \vdots & & \\ s_{i,j}T & & \\ \vdots & & \\ s_{M,1}T & \cdots & s_{M,M}T \end{bmatrix}$$

Actually, $\circ$ denotes the tensor product. Suppose A is of the form

$$A = T \circ E_{S,0} + E_{T,0} \circ S$$

where T and S are tridiagonal scaled M-matrices and $E_{S,0}$ and $E_{T,0}$ are diagonal matrices. For example, if

$$T = S = tridiag(-1/2, 1, -1/2),\ E_{T,0} = E_{S,0} = I$$

then A represents a central discretization of the Poisson equation on a square with Dirichlet boundary conditions.

Suppose T and S have the same order N, which is a power of 2. As in the previous section, we define the matrices T_i, S_i, $R_{T,i}$, $R_{S,i}$, $P_{T,i}$, $P_{S,i}$, $Q_{T,i}$, $Q_{S,i}$, $D_{T,i}$ and $D_{S,i}$. For any matrix $B = (b_{i,j})_{1 \le i,j \le N}$ let

$$rowsum(B) \equiv diag(\sum_{j=1}^{N} b_{i,j})_{1 \le i \le N}.$$

For $0 \leq i < n$, define

$$\begin{aligned}
E_T &\equiv rowsum(R_{T,i+1})E_{T,i} \cdot rowsum(P_{T,i+1}) \\
E_S &\equiv rowsum(R_{S,i+1})E_{S,i} \cdot rowsum(P_{S,i+1}) \\
P_{A,i+1} &\equiv P_{T,i+1} \circ P_{S,i+1} \\
R_{A,i+1} &\equiv R_{T,i+1} \circ R_{S,i+1} \\
D_{A,i+1} &\equiv D_{T,i+1} \circ D_{S,i+1} \\
E_{T,i+1} &\equiv D_{T,i+1}^{-1} E_T \\
E_{S,i+1} &\equiv D_{S,i+1}^{-1} E_S \\
Q_{A,i+1} &\equiv R_{T,i+1}T_iP_{T,i+1} \circ E_S + E_T \circ R_{S,i+1}S_iP_{S,i+1} \\
A_{i+1} &\equiv D_{A,i+1}^{-1} Q_{A,i+1} \\
&= T_{i+1} \circ E_{S,i+1} + E_{T,i+1} \circ S_{i+1}.
\end{aligned}$$

In the definition of TBF, we take

$$Q \equiv Q_{A,1},\ P \equiv P_{A,1},\ R \equiv R_{A,1}.$$

Note that if we would eliminate the word *rowsum* in the above definition, the TBF method would be direct, due to equation (3). Nevertheless, for smooth vectors, multiplication by a positive matrix B is well-approximated by multiplication by $rowsum\,(B)$. Since T_i is an M-matrix, $R_{T,i+1}$ and $P_{T,i+1}$ are positive. Consequently, if we use presmoothing and post-smoothing, i.e. $i_1 > 0$ and $i_2 > 0$, then the error is smooth, so equation (2) would give a good corrector for the current approximation. Moreover, the use of the above row-sum approximation makes the system (2) much easier to solve than the original system, since it includes 4 independent subsystems:

1. A diagonal system connecting variables which are odd in both directions, i.e., variables that correspond to odd rows of both S and T (fine grid system).

2. A tridiagonal system connecting variables which are odd in the first direction and even in the second, i.e., variables that correspond to odd rows of T and even rows of S (half coarse system).

3. A tridiagonal system connecting variables which are even in the first direction and odd in the second, i.e., variables that correspond to even rows of T and odd rows of S (half coarse system).

4. A penta-diagonal system connecting variables which are even in both directions, i.e., variables that correspond to even rows of both S and T (coarse grid system).

Only the solution of the last subsystem is expensive. The MBF method solves this subsystem recursively by the same procedure. On the i^{th} call to MBF $(0 < i \leq n)$ the operators used are

$$A \equiv A_{i-1},\ Q \equiv Q_{A,i},\ D \equiv D_{A,i},\ P \equiv P_{A,i} \text{ and } R \equiv R_{A,i}.$$

Since A_n is diagonal, the method is well-defined. The total work of MBF for a problem with N^2 variables is

$$w(N^2) = O(N^2) + w(N^2/4) = O(N^2) + O(N^2/4) + w(N^2/16) = \cdots = O(N^2)$$

Note that if we had

$$\begin{aligned}
(rowsum(R_{T,i+1}) \cdot rowsum(P_{T,i+1})) \circ I &= I \circ D_{S,i+1}^{-1} \\
I \circ (rowsum(R_{S,i+1}) \cdot rowsum(P_{S,i+1})) &= D_{T,i+1}^{-1} \circ I
\end{aligned}$$

then

$$\begin{aligned} Q_{A,i+1} &= D_{T,i+1}T_{i+1} \circ E_S + E_T \circ D_{S,i+1}S_{i+1} \\ &= T_{i+1} \circ E_{S,i} + E_{T,i} \circ S_{i+1} \end{aligned}$$

is already in the scaled form, on which the MBF algorithm may act recursively. Hence the scaling by D^{-1} in action (4) of Section 2.2 is not needed. Of course, these equalities cannot hold exactly, but if they hold approximately, we can avoid scaling. Especially in the non-separable case, where scaling is impossible, action (4) has to take place without scaling by D^{-1} (instead of actual scaling, we would prefer in that case to keep diagonal matrices multiplying the difference operators in each of the space directions). Hence we would like to assume the above equalities at least for all former levels, that is, at the i^{th} call to MBF, for all $0 \le j \le i-2$. We call that variant "noscal".

If we use the rectangular matrices P and R of the last part of Section 2.3, we get a variant of MBF which we call variant *notri*. In this variant, only the last of the four subsystems described above is solved. It assumes that the other tridiagonal subsystems affect only the smoothness of the error. The row-sum operation, however, is still performed on the original triangular matrices and not on the newly defined rectangular matrices.

Instead of the operators A_i and $Q_{A,i}$ defined above, one may use difference operators which arise from the original PDE. If the algorithm is to be automatic, all such operators have to be of the same type (i.e. central or upwind) as the original fine-grid operator. (Nevertheless, in Section 3 we will see that for some non-symmetric problems this condition has to be violated for the sake of convergence.) We use this strategy with the rectangular matrices of variant *notri*; our version is then different from classical multigrid only in the choice of restriction and prolongation operators. Note that the row-sums computed in the MBF algorithm are usually 4. Instead of the multiplication by these row-sums, one may divide the residual by 4 before action (4) of Section 2.2. Then one gets an algorithm which is equivalent to that of [12] for the Poisson equation, and is close to that of [5] for other problems. We denote that strategy MGF (Multigrid + Factorization). It should be kept in mind that when applying this strategy one must use $2^n - 1$ grid points on the finest grid and $2^q - 1$, $1 \le q < n$ for coarser grids in order to conserve uniformity. Here the even points, which are taken as coarse grid points, are always internal points of the original stencil. For 2^q point grids, on the other hand, the last fine grid point appears as a last grid point in all grids. Hence, coarse grids are biased towards the boundary. For our method MBF, on the other hand, stencils of both 2^n points or $2^n - 1$ points may be used. This is critical for implementation to problems on general regions, where grid lines may contain variable numbers of grid points (see Section 4).

As mentioned above, the MGF method requires division of residuals by 4 before action (4) takes place. Sometimes it is better to scale the discrete operators on all grids instead of dividing the residuals by this factor. Actually, for the Poisson equation both manners are equivalent: suppose A_1 has a coarse step-size $H = 2h$; then normalizing A_1 to have the same diagonal entries as $A_0 = A$ amounts to multiplication of A_1 by the factor 4, which is equivalent to the division of the residual in action (4) by that factor. Nevertheless, for differential equations that include derivatives of orders other than 2, this variant is not equivalent to MGF. We call it MGN (Multigrid + Normalization).

The generalization of the MBF method and of the other multigrid versions to nonseparable problems is straightforward. A tensor product by an $N \times N$ diagonal matrix is to be replaced with a multiplication by an $N^2 \times N^2$ diagonal matrix.

Another generalization of MBF is to non-rectangular domains. This is also straightforward, since a line containing an odd number of points may be divided into two sets, one containing odd points and the other containing even points. Then one of those sets is considered as a coarse grid, and is divided recursively in the same way. A similar strategy may be used in the other space direction.

3 NUMERICAL EXPERIMENTS

In this section, the MBF method is compared to other multigrid versions. The problems solved are of the type

$$\begin{aligned} Lu(x,y) &= f(x,y) \quad (x,y) \in (0,1)^2 \\ u(x,y) &= xy \qquad (x,y) \in \partial[0,1]^2 \end{aligned}$$

The equations are discretized via a 3-point central difference scheme. For MBF, the number of grid points in each space direction is $N = 2^n$. For other multigrid strategies, however, the number of grid points in each space direction is $N-1 = 2^n - 1$; otherwise, coarse grids are biased towards the boundary (see Section 2.4), and the convergence is slow.

For MBF, we have used variant "*notri*" of Section 2.4, in which tridiagonal subsystems are not solved. The main variant, which involves the solution of those subsystems, was found to be at most as effective as "*notri*".

The main variant of Section 2.4 was used for the hyperbolic, the non-uniform, the strongly indefinite and the discontinuous problems (the latter when applied with a red-black smoother). For other problems we have found that variant "noscal" described there (in which scaling of coarse-grid operators is omited), performs equally well. Hence we have chosen to use this simpler version rather than the main variant. Indeed, it was found that for most problems its performance was very close to that of the main variant. For the hyperbolic problem, however, its performance was twice as slow.

The smoother of the error in all grids was the one provided by the $ILU(1,1)$ iteration of [24] [25] [26] or the red-black (RB) iteration. This determines the operators Z_i of Section 2.2 to be the preconditioners for the $ILU(1,1)$ or RB iteration, respectively. These smoothers were found to be superior to the Jacobi and damped-Jacobi smoothers. One presmoothing and one post-smoothing is performed. The initial guess is random. Double precision arithmetic is used.

The integers in the following tables present the number of iterations needed to reduce the l_2 norm of the residual by 10^6. The maximum norm of the error was also computed, and its rate of convergence was close to that of the residual.

In conjunction with the MBF iteration, we have used the computer code of [23] that implements the vector acceleration RRE that was mentioned in the introduction. The RRE acceleration was employed in cycling mode, by restarting it after every 10 iterations until convergence. The results of this are compared to those provided by the MBF iteration without acceleration denoted by $NONE$.

We have also examined the classical multigrid versions mentioned at the end of Section 2.4. This strategy is denoted by the superscript ^{D}. The number of grid points in each space direction is $N-1 = 2^n - 1$. In most of the problems, the MGF and MGN versions of Section 2.4 are equivalent. Where this is not the case, we mention explicitly which of the two versions has been employed.

For comparison, we also checked the performance of a method which does not involve any multigrid strategy. This method is the Modified ILU method of [29] with the optimal parameter of [30], used as a preconditioner for RRE. It is denoted by $MILU$.

In the following tables, when a method converges very slowly we denote it by "slow", and when a method diverges, we denote it by "div".

3.1 The Poisson Equation

In table 1 we present the results for the Poisson equation.

	RRE	NONE	RRE	NONE	RRE	NONE	RRE	NONE	RRE
N	ILU	ILU	RB	RB	ILU^D	ILU^D	RB^D	RB^D	MILU
32	4	5	5	7	4	5	5	7	19
64	4	5	5	7	4	5	5	7	27
128	4	5	5	7	4	5	5	7	42

Table 1: Results for the Poisson equation

MBF and the classical multigrid version perform equally well for this problem.

3.2 Poisson Equation on a Tchebycheff-Type Grid

In table 2 we present the results for the Poisson equation, discretized via central differences on the 2-dimensional grid

$$P(j,k) \equiv \left(\frac{1-\cos(\frac{j\pi}{N+1})}{2}, \frac{1-\cos(\frac{k\pi}{N+1})}{2} \right) \quad 1 \le j,k \le N$$

The matrix operator for this scheme may be used as a preconditioner for a Tchebycheff-collocation discretization of the Poisson equation (see [31] and the references therein).

	RRE	NONE	RRE	NONE	RRE	NONE	RRE	NONE	RRE
N	ILU	ILU	RB	RB	ILU^D	ILU^D	RB^D	RB^D	MILU
32	4	5	7	10	4	5	8	11	16
64	4	6	9	19	5	6	10	18	23
128	5	7	13	33	5	6	14	28	36

Table 2: The Poisson equation with non-uniform grid

The superscript ^{D} refers to the MGF method of the end of Section 2.4, which is in the spirit of Dendy [12]. It performs equally well as MBF.

3.3 An Anisotropic Discontinuous Equation

In table 3 we present the results for an anisotropic equation whose coefficients are discontinuous;

$$a(x)u_{xx} + a(y)u_{yy} = 0$$

Here $a(t)$ is defined by

$$a(t) = \begin{cases} 0.01 & 0 < t \le 0.5 \\ 1 & 0.5 < t \le 1 \end{cases}$$

MBF and MGF perform equally well for this problem. For both methods, $N-1$ grid points were used in each space direction. If $N = 2^n$ grid points are used in any of the space directions, then for coarse-grid problems the discontinuity lines are biased towards the boundary, and convergence becomes slow.

Results similar to those of table 3 were obtained for the continues anisotropic problem

$$u_{xx} + 0.01u_{yy} = 0$$

This time, however, there was no difference in convergence rate when the number of grid points in each space direction was changed from $N = 2^n$ to $N-1 = 2^n - 1$, for both MBF and MGF.

	RRE	NONE	RRE	NONE	RRE	NONE	RRE	NONE	RRE
N	ILU	ILU	RB	RB	ILU^D	ILU^D	RB^D	RB^D	MILU
32	5	6	19	slow	5	6	19	slow	23
64	7	10	26	slow	7	10	26	slow	32
128	9	13	28	slow	9	13	28	slow	48

Table 3: An anisotropic discontinuous equation

3.4 A Convection-Diffusion Equation with Circular Streamlines

In table 4 we present the results for the convection-diffusion equation

$$u_{xx} + u_{yy} + 150(\ (y - 0.5)u_x - (x - 0.5)u_y) = f$$

whose characteristics are circles:

	RRE	NONE	RRE	NONE	RRE	NONE	RRE	NONE	RRE
N	ILU	ILU	RB	RB	ILU^D	ILU^D	RB^D	RB^D	MILU
32	7	10	12	15	8	slow	div	div	28
64	6	10	9	12	8	slow	div	div	51
128	6	10	9	12	8	slow	div	div	94

Table 4: A convection-diffusion equation with circular streamlines

Problems of the last type are widely discussed in [6]. The approach developed there requires special treatments and is not as automatic as ours.

3.5 A Convection-Diffusion Equation with Radial Streamlines

In table 5 we present the results for the convection-diffusion equation

$$u_{xx} + u_{yy} + 150(\ xu_x + yu_y) = f$$

whose characteristics are radial lines:

	RRE	NONE	RRE	NONE	RRE	NONE	RRE	NONE	RRE
N	ILU	ILU	RB	RB	ILU^D	ILU^D	RB^D	RB^D	MILU
32	7	9	slow	div	7	9	div	div	28
64	5	6	13	12	7	8	15	15	51
128	5	6	9	10	8	9	12	15	94

Table 5: A convection-diffusion equation with radial streamlines

The ^{D} superscript denotes here the MGF method of the end of Section 2.4. Nevertheless, the purely automatic MGF version, in which all difference operators are central, diverged. To avoid that we had to use upwind difference schemes for all grids coarser than the original grid. This strategy, however, though performing almost equally well as MBF, suffers the disadvantage of not being automatic. Another way to overcome divergence is to use the MGN method of the end of Section 2.4, with the same step-size h for all grids. This strategy is non-automatic as well, and about twice as slow as the first one.

3.6 A Convection Equation

In table 6 we present the results for the convection equation

$$(y-0.5)u_x-(x-0.5)u_y=f$$

whose characteristics are circles, discretized via an upwind scheme:

	RRE	*NONE*	*RRE*	*NONE*
N	*ILU*	*ILU*	ILU^D	ILU^D
32	10	27	10	slow
64	15	49	14	slow
128	23	89	23	slow

Table 6: A convection equation with circular streamlines

The superscript ^{D} refers to the *MGF* method of the end of Section 2.4. Its performance is equal to that of *MBF*.

The standard *ILU* and the Modified *ILU* of [29] (on the fine grid only, without multigrid strategy) do not converge for this problem. All multigrid strategies with an *RB* smoother are very slow.

3.7 The Helmholtz Equation

In table 7 we present the results for the Helmholtz equation

$$u_{xx}+u_{yy}+\beta u=f$$

with $\beta=64$. The *RRE* method for *MBF* was restarted in this example after every 5 iterations.

	RRE	*RRE*	*RRE*	*RRE*
N	*ILU*	*RB*	ILU^D	RB^D
32	9	14	17	16
64	9	13	17	19
128	8	15	18	20

Table 7: The Helmholtz equation

Without acceleration, all methods diverged. The *RRE* acceleration for *ILU* and *MILU* iterations (on the fine grid only, without multigrid strategy) also diverged.

The superscript ^{D} denotes here the *MGN* version, used with a continuation strategy; that is, use a parameter β smaller than that of the original PDE for grids coarser than the original grid, in such a way that the number $h^2\beta$ is constant for all grids. Without this continuation strategy, divergence was reported. Consequently, it suffers the disadvantage of not being automatic.

This problem is of the type of problems discussed in [7]. The projection approach given there requires more work and special treatment.

For $\beta>64$, the *RRE* acceleration for *MBF* seems to suffer stability problems, as the residual no longer decreases monotonically. A machine with higher precision (or Kacmarz smoother as in Section 3.8) is required. With classical multigrid, on the other hand, the acceleration is more stable: with the *ILU* smoother, it converges for $\beta=100$ and $N=64$ in 42 iterations.

Note that the Helmholtz equation and the convection-diffusion equations are better-posed as the number of grid points increase; hence the number of iterations generally decrease.

3.8 Helmholtz Equation with Mixed Boundary Conditions

The above experiments involve Dirichlet boundary conditions. In this sub-section we examine the Helmholtz equation with Dirichlet boundary conditions on the edges $x = 0$, $x = 1$ and $y = 1$, and with the mixed boundary conditions

$$\frac{\partial u}{\partial n} + \alpha u = g \tag{5}$$

on the edge $y = 0$.

We have repeated experiments 6.1 and 6.3 of [32], for which $\beta = 100$, $\alpha = 100i$, $N = 31$ and $\beta = 200$, $\alpha = 10i$, $N = 63$ respectively. On coarse grids, where the problem is very indefinite, we have used Kacmarz relaxations as a smoother. The cost of an MBF or MGF iteration was about 10 Jacobi iterations of the original problem. With MBF, we have converged in 10 iterations for the first problem and in 18 for the second one, which is much better than the results of [32]. With MGF, applied with the continuation strategy described in Section 3.7, we have converged in 23 iterations for the first problem.

3.9 Helmholtz Equation with Finite Elements

Finally, we have examined the Helmholtz equation

$$u_{xx} + u_{yy} + \beta u = f$$

with $\beta = 200$ and Dirichlet boundary conditions, discretized via bilinear finite elements. The grid for those elements is not uniform; in each space direction, the domain is divided into 4 elements, and the grid points are the roots of the Legendre polynomial of degree 17 in each element. Hence the total number of grid points is 63^2. This grid induces a division of the domain into squares, which serve as the bilinear finite elements. The matrix operator for this bilinear element scheme may be used as a preconditioner for a spectral element discretization of the Helmholtz equation (see [31] and the references therein). Though the coefficient matrix has nine non-zero diagonals, the operators for coarser grids have five non-zero diagonals only; they are obtained from the above finite difference approximation in the automatic or classical manners. Actually, this is a double discretization strategy. The relaxations on the finest grid are the ILU iteration or the four-color Gauss-Seidel iteration. On the second grid, the relaxation is ILU or RB iteration. One presmoothing and one post-smoothing are performed on those two grids. On coarser grids, since the operators are more indefinite, these relaxation methods are too divergent; hence, we use instead the Kacmarz iteration, 40 presmoothings and 40 post-smoothings on each level. Since on the third grid the number of points is 1/16 of that of the original grid, the total work on that grid is about five Jacobi relaxations of the original system. The cost of the whole multigrid or MBF procedure is about 10 such relaxations. RRE acceleration is restarted after every 10 multigrid or MBF iterations. The number of MBF iterations needed to reduce the residual by 6 orders of magnitude is 28 when ILU is used on the two finest grids and 27 when the Gauss Seidel smoother is used there. For MGF (with ILU on the two finest grids and with the continuation strategy of Section 3.7), the number of iterations needed is 52. When the residual is reduced by 6 orders of magnitude, the error is reduced by 6 orders for MBF and 5 orders for MGF.

We also examined the mixed boundary conditions case. For the mixed boundary conditions (5) on the edges $x = 0$ and $y = 0$, with $\beta = 200$, $\alpha = 10i$ and $N = 64$, MBF converged in 52 iterations, each costs about as much as 7 Jacobi iterations, with RRE restarted after every 20 iterations. Classical MG methods did not converge for this problem, even with the continuation strategy of Section 3.7.

3.10 Helmholtz Equation with Spectral Elements

The above double discretization strategy is not limited to bilinear element schemes; it was employed successfully for the spectral element scheme of [33] as well. Again, we relax the original equation on the finest grid; then we compute the difference scheme based on the nodes of the spectral elements, and use it to generate the coarse grid operators via MBF. These operators are used to find the coarse grid correction. For the Helmholtz equation with the mixed boundary conditions (5) on the edges $x = 0$ and $y = 0$, with the parameters $\beta = 100$, $\alpha = 100i$ and $N = 16$ and with 4×4 Legendre-type spectral elements, MBF converged in 16 iterations, each costs about as much as 5 Jacobi iterations (with RRE restarted after 10 iterations). For the same problem but with the parameters $\alpha = 200$, $\beta = 10i$ and $N = 64$, MBF converged in 60 iterations; each costs about as much as 2 Jacobi iterations. This rate of convergence was much better than that of the algorithm of [31], in which the spectral element scheme is preconditioned by a finite element or a finite difference scheme (even when the preconditioning system is solved by MBF). As a matter of fact, even for the Poisson equation our strategy was about 3 times faster than that of [31].

4 DISCUSSION

The MBF method is a version of multigrid, which is automatic in the sense that it depends on the algebraic system of equations rather than on the original PDE. Actually, it is a "black box" method for the solution of the linear system of equations. Hence it seems to be more robust than the classical multigrid method. For instance, nonsymmetric terms in the equation do not slow down the convergence, whether the characteristics are closed or open. Non-uniform grids are handled with the same efficiency, and no special treatment of the neighborhood of the boundary is needed. Moreover, when the RRE acceleration is applied to the method, it copes with the indefinite Helmholtz equation as well. For all these examples the rate of convergence is rather independent of the size of the problem. For anisotropic or pure advection problems, however, the rate of convergence of the MBF method applied with the RRE acceleration slightly depends on the size of the problem.

The MBF method is especially suited to use with the ILU smoother. The red-black smoother gives slightly worse results. The doubled damped Jacobi iteration as a smoother (with a damping parameter 0.5) was examined too. For all the above problems but the discontinuous-anisotropic and the hyperbolic problems, its performance was about twice slower than that of the ILU smoother. For those two problems, the damped Jacobi smoother was unsatisfactory.

The versions of multigrid denoted MGF and MGN perform well for problems which do not involve central first derivatives (including discontinuous and anisotropic problems). For problems which do contain central first derivatives, since the algorithm is assumed automatic, the discretization on coarse grids is of the same type as that of the fine grid, i.e. central. Hence divergence is often caused by the coarse-grids corrections. This difficulty can be handled by the special treatments of Section 3.5, but then the algorithm is no longer automatic. For the Helmholtz equation, one may overcome this difficulty by using a continuation strategy in the MGN version. Even though this (non-automatic) strategy is a bit slower than MBF, it is more stable and is applicable to more singular problems. If one uses Kacmarz relaxation on coarse grids, both MBF and MGF converge even for very indefinite problems, the MBF again being faster.

As opposed to the classical multigrid versions, the MBF is applicable whether the number of grid points in each space direction is even or odd. This indicates that it is applicable to problems defined on general regions. Given a region $\Omega \subset R^2$, one takes as a fine grid the restriction of an infinite 2-dimensional fine grid to Ω. For a coarser grid, one takes every other point (in both x and y space directions) in the infinite fine grid, and takes the restriction to Ω. The other coarse grids are created in the same way. As we have seen, MBF is not affected by the possibility that some coarse grid points lie near $\partial\Omega$. The coarse grid operators are created automatically as in the above description; this can be done easily by modifying the block sizes in the coefficient matrix of the system. Thus the algorithm is easy to program.

References

[1] Brandt, A.: Guide to Multigrid Development, Multigrid Methods. Hackbusch, W.; and Trottenberg, U., eds.: *Lecture Notes in Mathematics* 960, Springer-Verlag, Berlin, Heidelberg 1982.

[2] Hackbusch, W.: Multigrid Methods and Applications. Springer-Verlag, Berlin, Heidelberg, 1985.

[3] Hackbusch, W.: The Frequency decomposition Multigrid Algorithm. In Hackbusch, W., ed.: *Robust Multigrid Methods.* Proceedings of the fourth GAMM seminar. Kiel, January 1988.

[4] Alcouffe, R.; Brandt, A.; Dendy, J. E.; and Painter J.: The Multigrid Method for the Diffusion Equation with Strongly Discontinuous Coefficients. *SIAM J. Sci. Stat. Comp.*, vol. 2, 1981, pp. 430-454.

[5] Kettler, R.: Analysis and Comparison of Relaxation Schemes in Robust Multigrid and Preconditioned Conjugate Gradients Methods. In Hackbusch, W.; and Trottenberg, U., eds.: *Multigrid Methods, eds., Lecture Notes in Mathematics* 960, Springer-Verlag, Berlin, Heidelberg 1982.

[6] Yavneh, I.: Doctoral Thesis, Weizmann Institute of Science, Rehovot, Israel.

[7] Brandt, A.; and Ta'asan, S.: Multigrid methods for nearly singular and slightly indefinite problems. in *Lecture Notes in Mathematics* 960, Springer-Verlag, Oct. 1985, pp. 100-122.

[8] Brandt, A.: Algebraic Multigrid Theory: The Symmetric Case. *Appl. Math. Comp.*, vol. 19, 1986, pp. 23-56.

[9] Brandt, A.; McCormick, S. F.; and Ruge, J.: Algebraic Multigrid (AMG) for Automatic Multigrid Solution with Application to Geodetic Computation. *SIAM J. Sci. Stat. Comp.*

[10] Ruge, J. W.; and Stuben, K.: Algebraic Multigrid. In McCormick, S.F., ed.: *Multigrid Methods*, Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania, 1987.

[11] McCormick, S. F.: An Algebraic Interpretation of Multigrid Methods. *SIAM. J. Num. Anal.*, vol. 19, 1982, pp. 548-560.

[12] Dendy, J. E.: Black Box Multigrid. *J. Comp. Phys.*, vol. 48, 1982, pp. 366-386.

[13] Dendy, J. E.: Black Box Multigrid for Nonsymmetric Problems. *Appl. Math. Comp.*, vol. 13, 1983, pp. 261-283.

[14] Dendy, J. E.: Black Box Multigrid for Periodic and Singular Problems. Los Alamos National Laboratories, LA-UR-86-3897.

[15] Dendy, J. E.: A Multigrid Method for 3-D Problems with Discontinuous Coefficients. Los Alamos National Laboratories, LA-UR-86-1165.

[16] Dendy, J. E.: Black Box Multigrid for Systems. *Appl. Math. Comp.*, vol. 19, 1986.

[17] Eddy, R. P.: Extrapolating to the limit of a vector sequence. in Wang, P. C. C. ed: *Information Linkage Between Applied Mathematics and Industry.* Academic Press, N. Y., 1979, pp. 387-396.

[18] Mesina, M.: Convergence acceleration for the iterative solution of the equations $X = AX + f$. *Comp. Meth. Appl. Mech. Eng.*, vol. 10, 1977, pp. 165-173.

[19] Smith, D. A.; Ford, W. F.; Sidi, A.: Extrapolation methods for Vector Sequences. *SIAM Rev.*, vol. 29, 1987, pp. 199-233.

[20] Sidi, A.: Convergence and Stability Properties of MPE and RRE algorithms. *SIAM J. Num. Anal.,* vol. 23, 1986, pp. 197-206.

[21] Sidi, A.; Bridger, J.: Convergence and Stability Analysis for Some Vector Extrapolation Methods in the Presence of Defective Iteration Matrices. *J. Comp. Appl. Math.,* vol. 22, 1988, pp. 35-61.

[22] Sidi, A.: Extrapolation vs. Projection Methods for Linear Systems of Equations. *J. Comp. Appl. Math.,* vol. 22, 1988, pp. 71-88.

[23] Sidi, A.: Efficient Implementation of Minimal Polynomial and Reduced Rank Extrapolation Methods. *J. Comp. Appl. Math.,* vol. 36, 1991, pp. 305-337.

[24] Meijerink, J. A.; Van der Vorst H. A.: An Iterative Solution Method for Linear Systems of Which the Coefficients Matrix is a Symmetric M-Matrix. *Math. Comp.,* vol. 31, 1977, pp. 148-162.

[25] Meijerink, J. A.; and Van der Vorst, H. A.: Guidelines for the Usage of Incomplete Decompositions in Solving Sets of Linear Equations as they Occur in Practical Problems. *J. Comp. Phys.,* vol. 44, 1981, pp. 134-155.

[26] Van der Vorst, H. A.: Iterative Solution Methods for Certain Sparse Linear Systems with a Non-symmetric Matrix Arising from PDE-problems. *J. Comp. Phys.,* vol. 49, 1982, pp. 1-19.

[27] Van der Vorst, H. A.: Preconditioned Tchebycheff iterative solution method for large sparse linear systems with a non-symmetric matrix. *Lecture Notes in Mathematics* 968, pp. 323-333.

[28] Van der Vorst, H. A.: High Performance Preconditioning. *SIAM J. Sci. Stat. Comp.,* vol. 10, 1989, pp. 1175-1184.

[29] Gustafsson, S.: On modified incomplete factorization methods. *Lecture Notes in Mathematics,* 968, pp. 334-351.

[30] Chan, T. F.: Fourier Analysis of Relaxed Incomplete Factorization Preconditioners. *SIAM J. Sci. Stat. Comp.,* vol. 12, 1991, pp. 668-680.

[31] Quarteroni, A.; and Zampieri, E.: Finite Element Preconditioning for Legendre Spectral Collocation Approximations to Elliptic Equation and Systems. *SIAM J. Num. Anal.,* vol. 29, 1992, pp. 917-936.

[32] Freund, R. W.: Conjugate gradients type methods for linear systems with complex symmetric coefficient matrices. *SIAM J. Sci. Stat. Comp.,* vol. 13, 1992, pp. 425-448.

[33] Patera, A. T.: A Spectral Element Method for Fluid Dynamics; Laminar Flow in a Channel Expansion. *J. Comp. Phys.,* vol. 54, no. 3, June 1984.

[34] Bastian, P.; and Horton, G.: Parallelization of Robust Multigrid Methods: ILU Factorization and Frequency Decomposition Method. *SIAM. J. Sci. Stat. Com.,* vol. 12, 1991, pp. 1457-1470.

[35] Varga, R.: *Matrix Iterative Analysis.* Prentice-Hall, N. J., 1962.